

PROGRAM STUDI TEKNIK ROBOTIKA
POLITEKNIK NEGERI BATAM

ROBOT PENGANTAR BARANG

LAPORAN AKHIR PROYEK

VERSION <1.0>

21/JANUARI/2020

Nama Proyek	Robot Pengantar Barang Untuk Mendukung Social Distancing Dalam Mencegah Penyebaran Covid-19		
Anggota Tim	1. Yeni Riska Pratiwi (4221901001) 2. Rahel Yulianti (4221901004) 3. Muh Rihad Afra Graha Pratama (4221901007) 4. Muh Rifalwan (4221901008) 5. Rici Rahman (4221901021)		
Project Manager	1. Senanjung Prayoga, S.Pd., MT 2. Hendawan Soebhakti, ST., MT		
Tanggal Mulai Proyek	19 Oktober 2020	Tanggal Selesai	27 Januari 2021

BAGIAN 1: RINGKASAN PROYEK

1.1 PERMASALAHAN DAN TUJUAN

1.1.1 Permasalahan

- Bagaimana cara mengendalikan pergerakan robot?
- Bagaimana robot mengetahui posisinya di lingkungan kerjanya?
- Bagaimana cara robot mendeteksi halangan yang ada di lintasan gerakannya?
- Bagaimana cara robot dapat mengisi baterai secara otomatis?
- Bagaimana cara menerima permintaan (order) dari customer?

1.1.2 Tujuan dan manfaat

1.1.2.1 Tujuan

- Membuat pergerakan robot menggunakan sistem akselerasi & dekselerasi
- Mengimplementasikan sistem indoor GPS pada robot
- Desain sensor ultrasonik sebagai sensor deteksi halangan pada robot
- Pemasangan fitur auto docking pada robot
- Pembuatan web based robot app

1.1.2.2 Manfaat

- a) Manfaat yang diharapkan dari hasil penelitian ini adalah mencegah penyebaran Covid-19

1.2 PEMBAGIAN TUGAS

No	Nama	Tugas dan Tanggungjawab	Target Pencapaian
1	Rahel Yulianti	Mengendalikan pergerakan robot menggunakan <i>inverse kinematic</i>	Membuat pergerakan motor pada robot menggunakan <i>inverse kinematic</i>
2	Yeni Riska Pratiwi	Membaca posisi robot berdasarkan data indoor GPS dan membuat perencanaan gerak robot point-to-point	Menampilkan peta posisi terakhir robot
3	Rici Rahman	Mendeteksi objek disekitar robot dengan sensor ultrasonik	Robot dapat mendeteksi halangan dan menghindarinya
4	Muhammad Rihad Afra Graha Pratama	Membaca kapasitas baterai robot dan melakukan pengisian baterai secara otomatis pada <i>docking station</i>	Robot Dapat melakukan pengisian baterai secara otomatis
5	Muhammad Rifalwan	Membaca data permintaan (order) dari customer dan merekam data jenis order dan posisi meja customer	Web dapat diakses oleh customer

1.3 TARGET YANG TELAH / BELUM DICAPAI

1.3.1 Tabel pencapaian

No	Nama	Sudah Dicapai	Belum Dicapai	Alasan
1	Rahel Yulianti	1. Test kecepatan motor dengan inverse kinematic 2. Kontrol robot menggunakan koneksi bluetooth via android 3. Menambahkan akselerasi & dekselerasi pada motor		
2	Yeni Riska Pratiwi	1. Instal driver IA indoor GPS 2. Menampilkan posisi terakhir mobile beacon & robot di dashboard		
3	Rici Rahman	1. Menyelesaikan prototype delivery robot 2. Sensor sudah dapat mendeteksi halangan		Masih dalam tahap pembelajaran
4	Muhammad Rihad Afra Graha Pratama	1. Wiring elektrik 2. Membuat skematik wiring robot	Desain docking station	Masih dalam tahap pembelajaran
5	Muhammad Rifalwan	1. Desain mekanik dari delivery robot 2. Menyelesaikan prototype delivery robot 3. Desain web aplikasi		

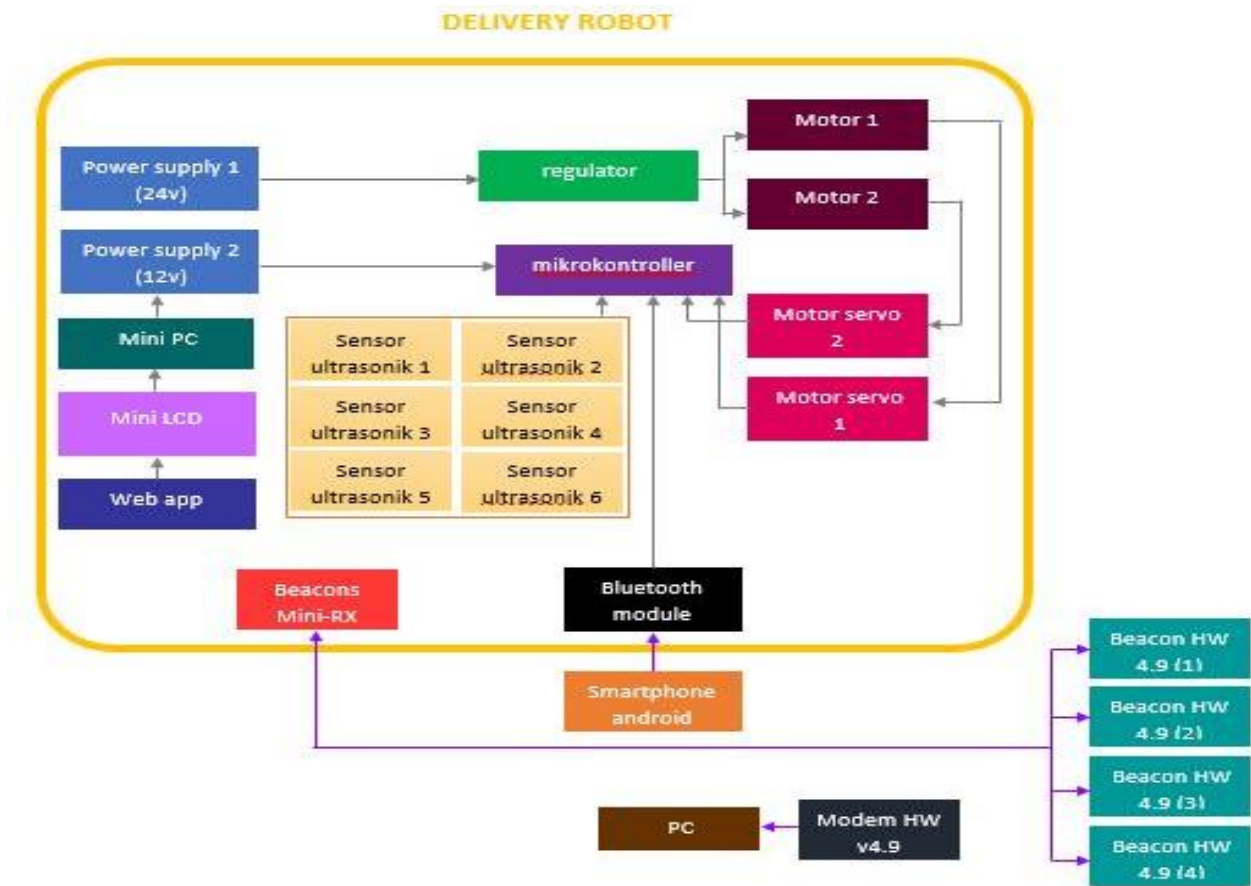
1.3.2 Tabel diagram waktu

No	PEKERJAAN	TANGGAL DAEDLINE	TANGGAL DIKERJAKAN	MASALAH
1	fabrikasi mekanik	19 - 23 Oktober	20 - 30 Oktober	material
2	wiring elektrikal	26 - 30 Oktober	3 -5 November	alat & bahan
3	uji coba sistem mekanik & elektrikal robot	2 -6 November	5 -15 November	tahap pembelajaran
4	mengikuti event v factor indonesia	18 - 20 November	18 - 20 November	
5	UTS	19 - 27 November	19 - 27 November	
6	integrasi sistem lokalisasi robot	30 - 4 Desember	1 - 14 januari	tahap pembelajaran
7	mengendalikan pergerakan motor dengan akselerasi & dekselerasi	27 desember - 25 januari	27 desember - 25 januari	
8	desain sistem deteksi halangan robot	27 desember - 25 januari	29 desember - 25 januari	
9	desain sistem pengisian baterai & docking	27 desember - 25 januari	28 desember - 25 januari	
10	desain web app (aplikasi pesanan customer robot)	30 desember - 25 januari	29 desember - 25 januari	
11	pemograman point to point	7 - 18 Desember	belum terlaksana	
12	pemasangan mini pc	21 - 25 Desember		
13	pemasangan LCD	28 - 31 Desember		
14	pemasangan speaker	4 - 8 Januari		
15	wiring keseluruhan	11 - 15 Januari		
16	pengujian seluruh sistem robot	18 - 29 Januari		
17	pembuatan dokumen desain industri	1 - 5 Februari		
18	penulisan papper & laporan akhir	8 - 12 Februari		

BAGIAN 3: GAMBARAN DESAIN

3.1 DESAIN SISTEM

Berikut desain sistem yang dibuat pada proyek dalam bentuk diagram blok .



Gambar 11. Diagram perangkat keras pada robot

GPS ultrasonik digunakan sebagai sistem navigasi robot untuk bergerak dari satu tempat ke tempat lainnya. Sebagai pusat kendali digunakan mini PC yang akan mengirimkan data gerakan robot ke mikrokontroler. Robot pengantar barang ini dirancang untuk dapat dipanggil menggunakan smartphone. Pada tahap awal ini, robot hanya bisa dipanggil dan mengantarkan barang dari tempat yang sudah diregister sebelumnya oleh sistem sehingga perlu dibuat sebuah aplikasi pada smart phone untuk memanggil robot dengan pilihan tempat penjemputan dan pengantaran.

Sebuah charging station diperlukan untuk melakukan pengisian baterai robot secara otomatis. Untuk dapat melakukan pengecasan ini, maka pada robot harus dipasang sistem monitoring baterai. Sensor jarak dengan ultrasonik dipasang pada bagian depan dan belakang robot untuk mendeteksi adanya halangan sehingga dapat dihindari atau berhenti.

Jenis controller board yang akan atau telah diimplementasi pada alat yang dikembangkan adalah :

1. Arduino mega (1 pcs)
2. Arduino uno (2 pcs)

Alasan memilih Arduino sebagai controller board adalah :

- Umum
 1. Selaras dengan mata kuliah aplikasi mikrokontroller
 2. Harga terjangkau
 3. Bahasa pemrograman Arduino mudah dipahami
 4. Bersifat open source
 5. Referensi mengenai Arduino cukup banyak ditemui
 6. Antar muka antara modul arduino dengan computer melalui port USB. Ada beberapa keuntungan dengan menggunakan antar muka USB, yaitu selain sebagai transfer data dari komputer ke modul arduino dan sebaliknya juga bisa menyediakan tegangan DC 5 volt untuk modul arduino tersebut.
- Spesifik (Arduino Mega 2560 Rev 3)
 1. Memiliki digital I/O yang cukup banyak
 2. Memiliki pin RX/TX lebih banyak (4 pin)
 3. memori EEPROM (Electrically Erasable Programmable Read Only Memory) lebih besar, 4KB.
 4. memori Flash (untuk diisi program) lebih besar, 256 KB

- Spesifik (arduino UNO Rev 3)

1. Mikrokontroler ini sudah disediakan pihak kampus (Tidak perlu membeli)
2. Menggunakan USB jack (Lebih mudah ditemukan)
3. Memiliki internal regulator dan fuse/sekring elektronik, sehingga lebih aman jika menggunakan sumber daya external
4. Operating input voltasenya bisa 12v (karena delivery robot menggunakan baterai 12v)

Komunikasi yang digunakan untuk menghubungkan system yang satu dengan yang lainnya pada delivery robot adalah Arduino mega yang dapat berkomunikasi dengan 2 pcs Arduino uno, kita membutuhkan komunikasi serial (komunikasi yang pengiriman datanya per-bit secara berurutan dan bergantian.) Komunikasi serial ini membutuhkan pin transmitter dan pin receiver dimana kita harus menghubungkan :

- RX dengan TX

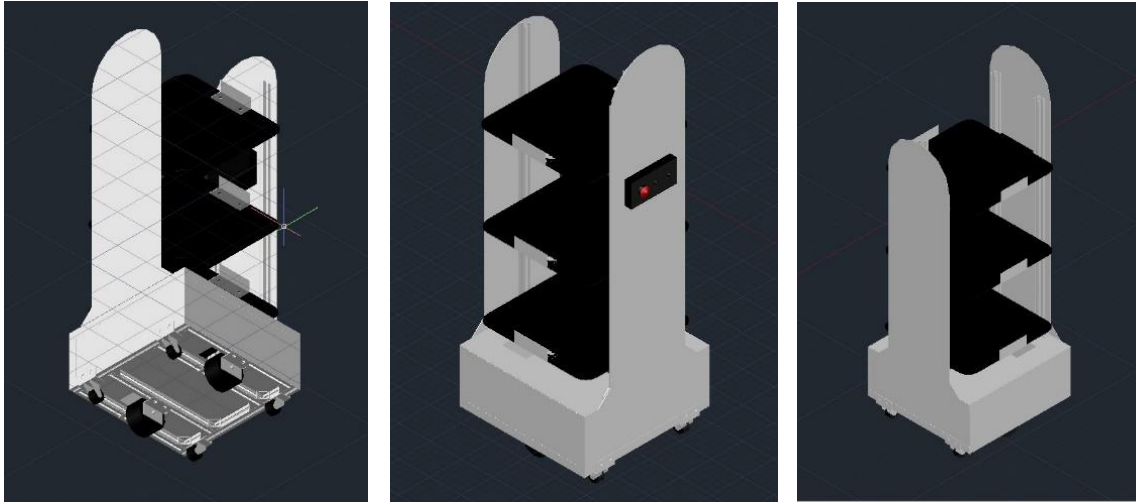
- TX dengan RX

Arduino UNO sebagai slave yang masing-masing arduino UNO dipasangkan ke driver motor. Arduino UNO berfungsi sebagai pengontrol kecepatan motor, data yang diberikan arduino ke driver berupa pulsa digital dengan interval delayMicroseconds. Digunakannya master dan slave karena jika seluruh program disatukan dalam satu arduino maka penulisan pulsa digital ke driver akan terganggu oleh proses yang lainnya, sehingga intervalnya tidak beraturan.

Arduino MEGA sebagai master dan pengolah data inverse kinematik, arah gerak motor, encoder, parsing data dan input output lainnya. Arduino mega mendapatkan inputan data melalui komunikasi bluetooth serial yang dihubungkan dengan ponsel android.

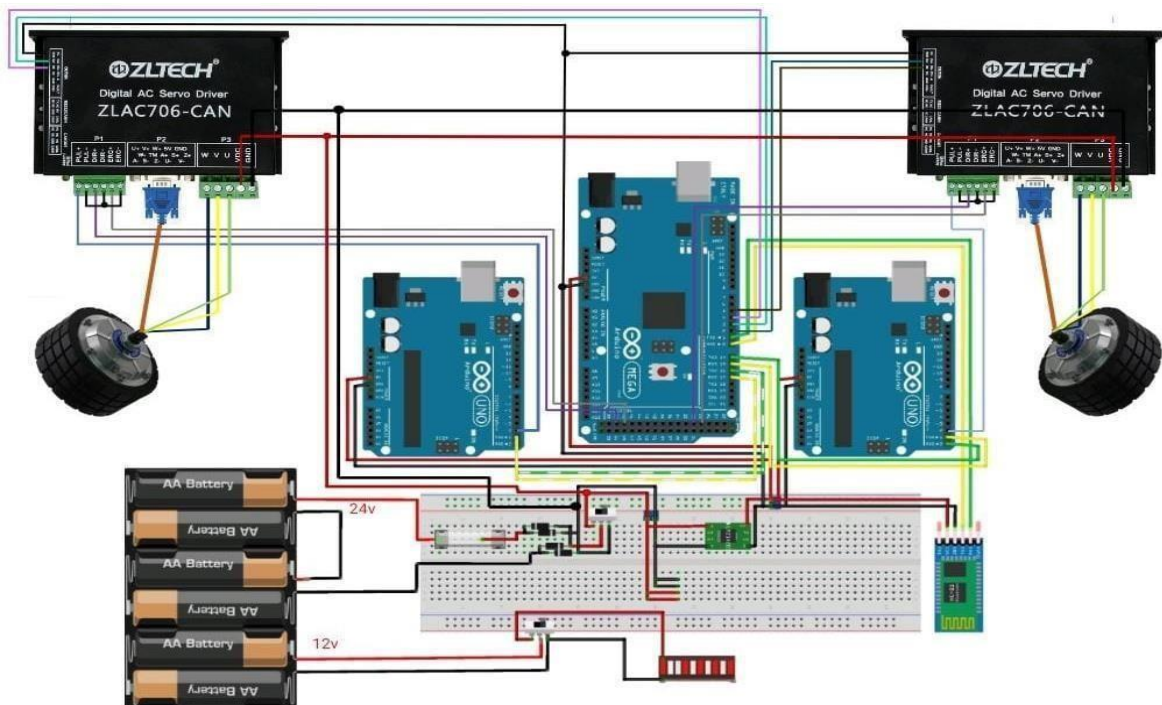
3.2 DESAIN HARDWARE

3.2.1 Gambar Mekanik



Gambar 12. Desain mekanik robot

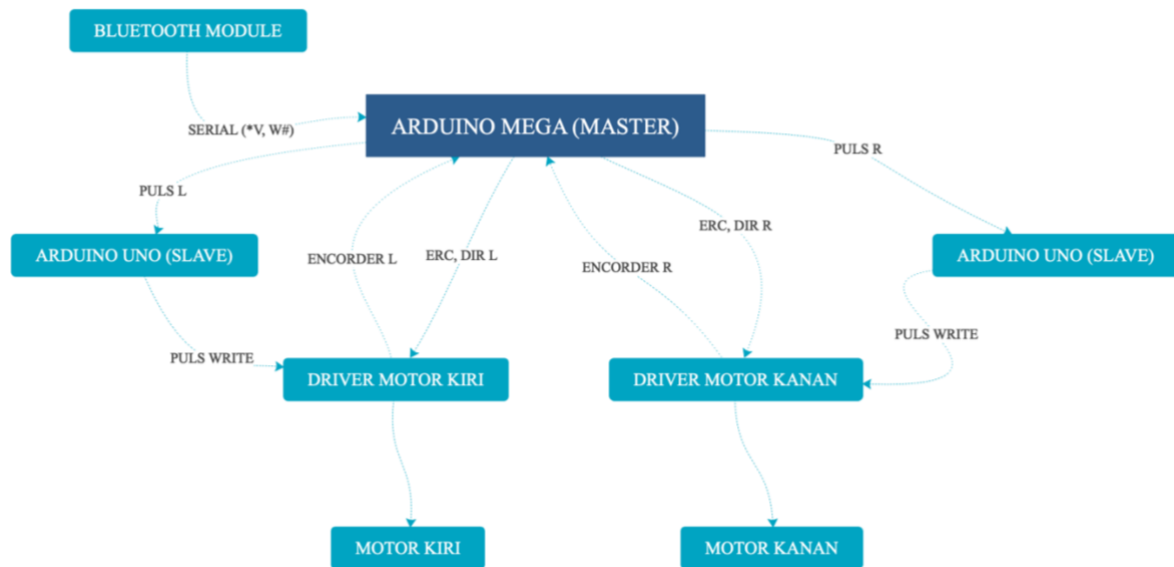
3.2.2 Gambar Rangkaian Elektronika



Gambar 13. Desain elektrikal robot

3.3 DESAIN SOFTWARE

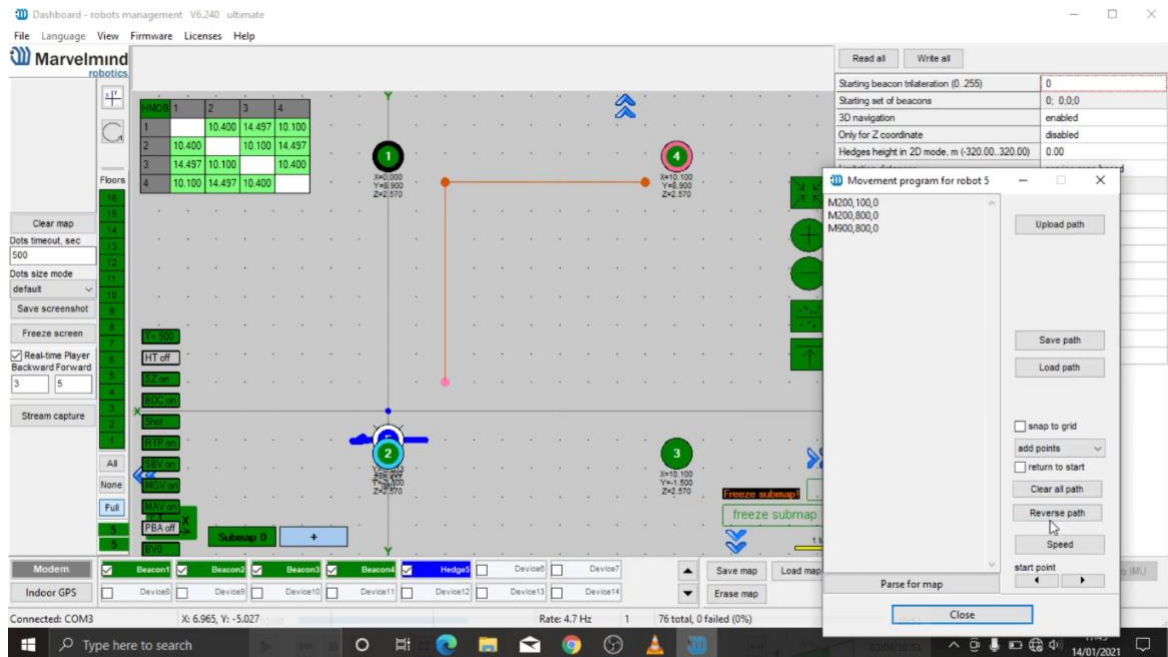
3.3.1 Skema Program Robot



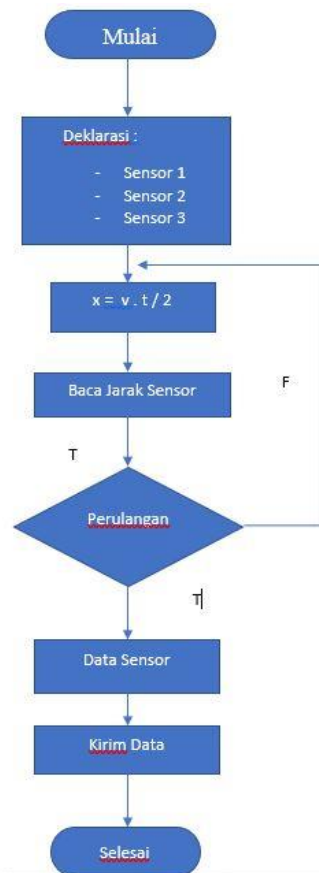
3.3.2 Skema Pergerakan Motor



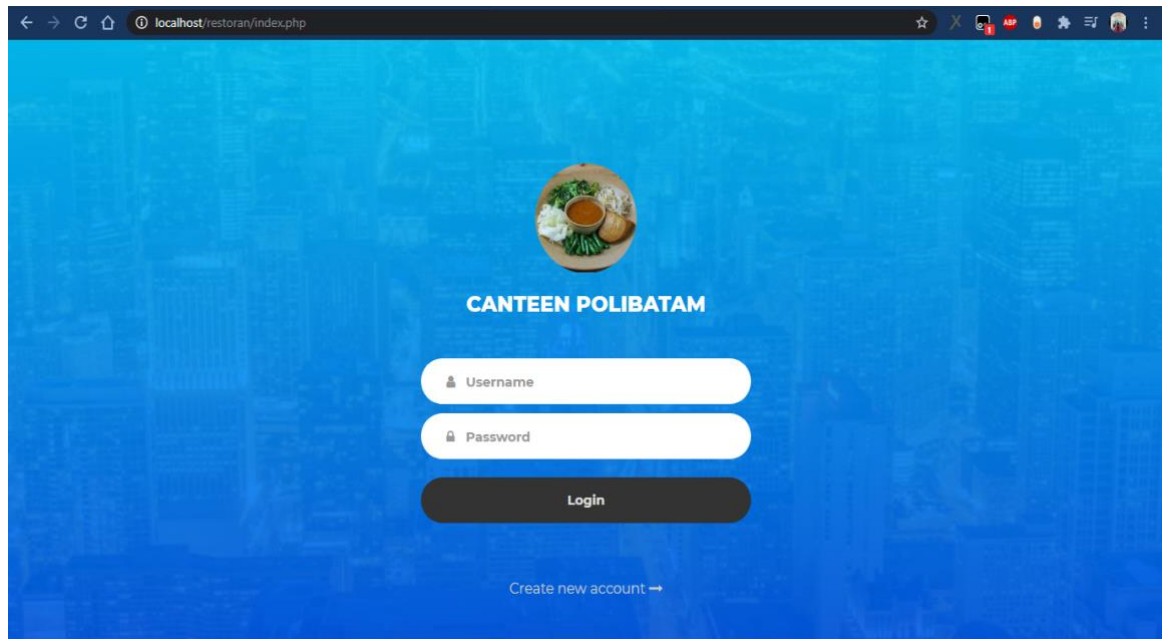
3.3.3 Skema Sistem Lokalisasi Robot



3.3.4 Skema Sistem Deteksi Halangan



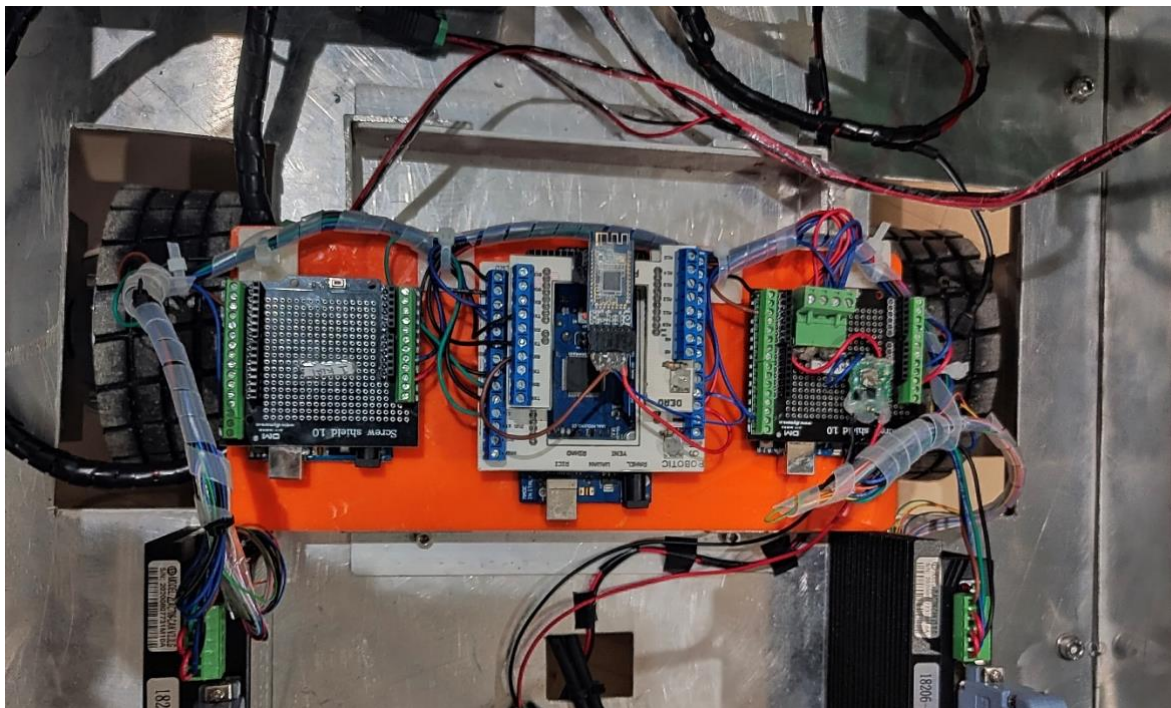
3.3.5 Skema Desain Web



BAGIAN 4: HASIL DAN PENGUJIAN



Gambar 14. Desain mekanik robot aktual



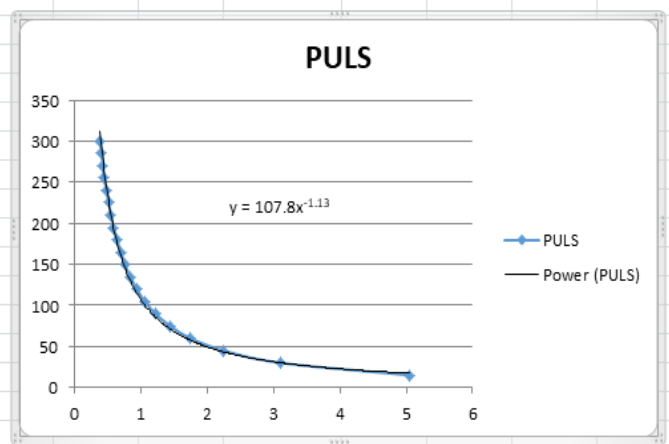
Gambar 15. Desain elektrik robot aktual

4.1 Sistem Roda Penggerak Robot

4.1.1 Analisa kecepatan motor

1. Parameter yang diuji : Kecepatan motor
2. Alat yang digunakan :
 - Motor & Driver
 - Baterai aki
 - Arduino
 - Laptop
3. Hasil regresi kecepatan motor

NO	RPS	PULS	Y	error
1	5.04	15	17.33294	2.33%
2	3.11	30	29.90884	-0.09%
3	2.25	45	43.1174	-1.88%
4	1.75	60	57.27773	-2.72%
5	1.44	75	71.39522	-3.60%
6	1.22	90	86.10575	-3.89%
7	1.06	105	100.9307	-4.07%
8	0.94	120	115.607	-4.39%
9	0.84	135	131.2753	-3.72%
10	0.76	150	146.9939	-3.01%
11	0.7	165	161.3088	-3.69%
12	0.64	180	178.4988	-1.50%
13	0.59	195	195.6843	0.68%
14	0.55	210	211.8405	1.84%
15	0.52	225	225.7018	0.70%
16	0.48	240	247.0678	7.07%
17	0.45	255	265.7594	10.76%
18	0.43	270	279.7689	9.77%
19	0.41	285	295.2385	10.24%
20	0.39	300	312.4034	12.40%



4.1.2 Analisa akselerasi dan dekselerasi pada motor

1. Parameter yang diuji : akselerasi dan dekselerasi pada motor
2. Alat yang digunakan :
 - Robot
 - Laptop

3. Hasil :

- Akselerasi

ml>0.00,0.00 l	ml>0.00,0.00	ml>78.30,819.20 l	ml>78.30,819.20
ml>0.00,0.00 l	ml>0.00,0.00	ml>78.30,655.36 l	ml>78.30,655.36
ml>0.00,0.00 l	ml>0.00,0.00	ml>78.30,655.36 l	ml>78.30,655.36
ml>0.00,0.00 l	ml>0.00,0.00	ml>78.30,524.29 l	ml>78.30,524.29
ml>0.00,0.00 l	ml>0.00,0.00	ml>78.30,524.29 l	ml>78.30,524.29
ml>0.00,0.00 l	ml>0.00,0.00	ml>78.30,419.43 l	ml>78.30,419.43
ml>0.00,0.00 l	ml>0.00,0.00	ml>78.30,419.43 l	ml>78.30,419.43
ml>0.00,0.00 l	ml>0.00,0.00	ml>78.30,335.54 l	ml>78.30,335.54
ml>0.00,0.00 l	ml>0.00,0.00	ml>78.30,335.54 l	ml>78.30,335.54
ml>0.00,0.00 l	ml>0.00,0.00	ml>78.30,268.44 l	ml>78.30,268.44
ml>0.00,0.00 l	ml>0.00,0.00	ml>78.30,268.44 l	ml>78.30,268.44
ml>0.00,0.00 l	ml>0.00,0.00	ml>78.30,214.75 l	ml>78.30,214.75
ml>0.00,0.00 l	ml>0.00,0.00	ml>78.30,214.75 l	ml>78.30,214.75
ml>78.30,2000.00 l	ml>78.30,2000.00	ml>78.30,171.80 l	ml>78.30,171.80
ml>78.30,2000.00 l	ml>78.30,2000.00	ml>78.30,171.80 l	ml>78.30,171.80
ml>78.30,1600.00 l	ml>78.30,1600.00	ml>78.30,137.44 l	ml>78.30,137.44
ml>78.30,1600.00 l	ml>78.30,1600.00	ml>78.30,137.44 l	ml>78.30,137.44
ml>78.30,1280.00 l	ml>78.30,1280.00	ml>78.30,109.95 l	ml>78.30,109.95
ml>78.30,1280.00 l	ml>78.30,1280.00	ml>78.30,109.95 l	ml>78.30,109.95
ml>78.30,1024.00 l	ml>78.30,1024.00	ml>78.30,87.96 l	ml>78.30,87.96
ml>78.30,1024.00 l	ml>78.30,1024.00	ml>78.30,87.96 l	ml>78.30,87.96
ml>78.30,819.20 l	ml>78.30,819.20	ml>78.30,70.37 l	ml>78.30,70.37
ml>78.30,819.20 l	ml>78.30,819.20	ml>78.30,70.37 l	ml>78.30,70.37
ml>78.30,655.36 l	ml>78.30,655.36	ml>78.30,70.37 l	ml>78.30,70.37
ml>78.30,655.36 l	ml>78.30,655.36	ml>78.30,70.37 l	ml>78.30,70.37
ml>78.30,524.29 l	ml>78.30,524.29	ml>78.30,70.37 l	ml>78.30,70.37
ml>78.30,524.29 l	ml>78.30,524.29	ml>78.30,70.37 l	ml>78.30,70.37
ml>78.30,419.43 l	ml>78.30,419.43	ml>78.30,70.37 l	ml>78.30,70.37
ml>78.30,419.43 l	ml>78.30,419.43	ml>78.30,70.37 l	ml>78.30,70.37
ml>78.30,335.54 l	ml>78.30,335.54	ml>78.30,70.37 l	ml>78.30,70.37
ml>78.30,335.54 l	ml>78.30,335.54	ml>78.30,70.37 l	ml>78.30,70.37
ml>78.30,268.44 l	ml>78.30,268.44	ml>78.30,70.37 l	ml>78.30,70.37
ml>78.30,268.44 l	ml>78.30,268.44	ml>78.30,70.37 l	ml>78.30,70.37
ml>78.30,214.75 l	ml>78.30	ml>78.30,70.37 l	ml>78.30

☒ Autoscroll

- Dekselerasi

m1>78.30,70.37	m2>78.30,70.37	m1>0.00,210.12	m2>0.00,210.12
m1>78.30,70.37	m2>78.30,70.37	m1>0.00,252.14	m2>0.00,252.14
m1>78.30,70.37	m2>78.30,70.37	m1>0.00,252.14	m2>0.00,252.14
m1>78.30,70.37	m2>78.30,70.37	m1>0.00,302.57	m2>0.00,302.57
m1>78.30,70.37	m2>78.30,70.37	m1>0.00,302.57	m2>0.00,302.57
m1>78.30,70.37	m2>78.30,70.37	m1>0.00,363.09	m2>0.00,363.09
m1>78.30,70.37	m2>78.30,70.37	m1>0.00,363.09	m2>0.00,363.09
m1>78.30,70.37	m2>78.30,70.37	m1>0.00,435.70	m2>0.00,435.70
m1>78.30,70.37	m2>78.30,70.37	m1>0.00,435.70	m2>0.00,435.70
m1>78.30,70.37	m2>78.30,70.37	m1>0.00,522.85	m2>0.00,522.85
m1>78.30,70.37	m2>78.30,70.37	m1>0.00,522.85	m2>0.00,522.85
m1>78.30,70.37	m2>78.30,70.37	m1>0.00,627.41	m2>0.00,627.41
m1>0.00,84.44	m2>0.00,84.44	m1>0.00,627.41	m2>0.00,627.41
m1>0.00,84.44	m2>0.00,84.44	m1>0.00,752.90	m2>0.00,752.90
m1>0.00,101.33	m2>0.00,101.33	m1>0.00,752.90	m2>0.00,752.90
m1>0.00,101.33	m2>0.00,101.33	m1>0.00,903.48	m2>0.00,903.48
m1>0.00,121.60	m2>0.00,121.60	m1>0.00,903.48	m2>0.00,903.48
m1>0.00,121.60	m2>0.00,121.60	m1>0.00,1084.17	m2>0.00,1084.17
m1>0.00,145.92	m2>0.00,145.92	m1>0.00,1084.17	m2>0.00,1084.17
m1>0.00,145.92	m2>0.00,145.92	m1>0.00,1301.01	m2>0.00,1301.01
m1>0.00,175.10	m2>0.00,175.10	m1>0.00,1301.01	m2>0.00,1301.01
m1>0.00,175.10	m2>0.00,175.10	m1>0.00,1561.21	m2>0.00,1561.21
m1>0.00,210.12	m2>0.00,210.12	m1>0.00,1561.21	m2>0.00,1561.21
m1>0.00,210.12	m2>0.00,210.12	m1>0.00,1873.45	m2>0.00,1873.45
m1>0.00,252.14	m2>0.00,252.14	m1>0.00,1873.45	m2>0.00,1873.45
m1>0.00,252.14	m2>0.00,252.14	m1>0.00,2248.14	m2>0.00,2248.14
m1>0.00,302.57	m2>0.00,302.57	m1>0.00,0.00	m2>0.00,0.00
m1>0.00,302.57	m2>0.00,302.57	m1>0.00,0.00	m2>0.00,0.00
m1>0.00,363.09	m2>0.00,363.09	m1>0.00,0.00	m2>0.00,0.00
m1>0.00,363.09	m2>0.00,363.09	m1>0.00,0.00	m2>0.00,0.00
m1>0.00,435.70	m2>0.00,435.70	m1>0.00,0.00	m2>0.00,0.00
m1>0.00,435.70	m2>0.00,435.70	m1>0.00,0.00	m2>0.00,0.00
m1>0.00,522.85	m2>0.00,522.85	m1>0.00,0.00	m2>0.00,0.00
m1>		m1	

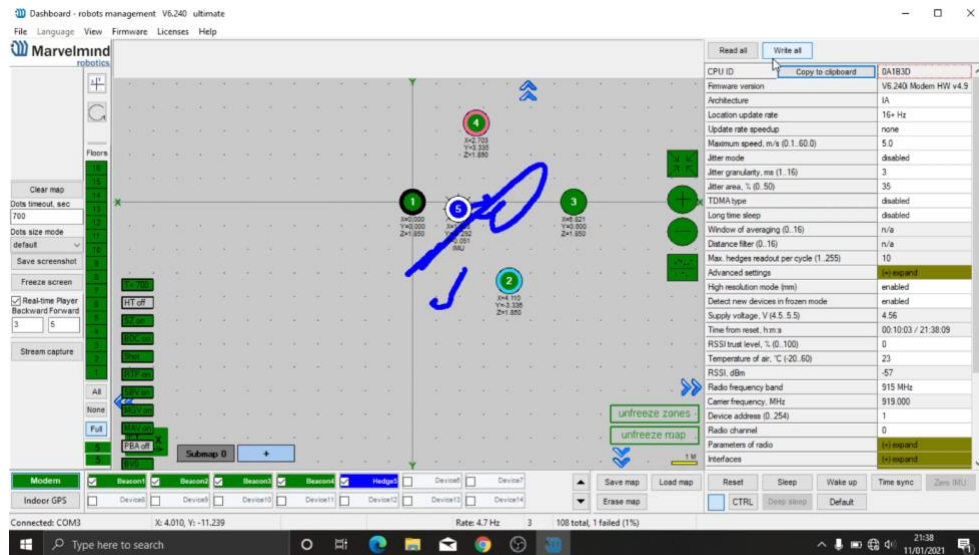
☒ Autocroll

☒ Autocroll

4.2 Sistem Lokalisasi Robot

4.2.1 Analisa Sistem Lokalisasi

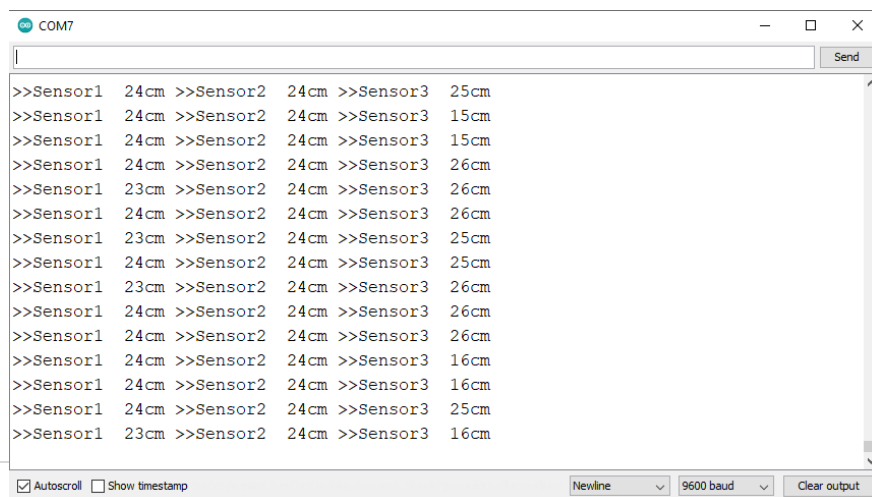
1. Parameter yang diuji : Lokalisasi robot
2. Alat yang digunakan :
 - Indoor GPS marvelmind robotics
 - robot
3. Hasil :



4.3 Sistem Deteksi Halangan

4.3.1 Analisa Deteksi Halangan

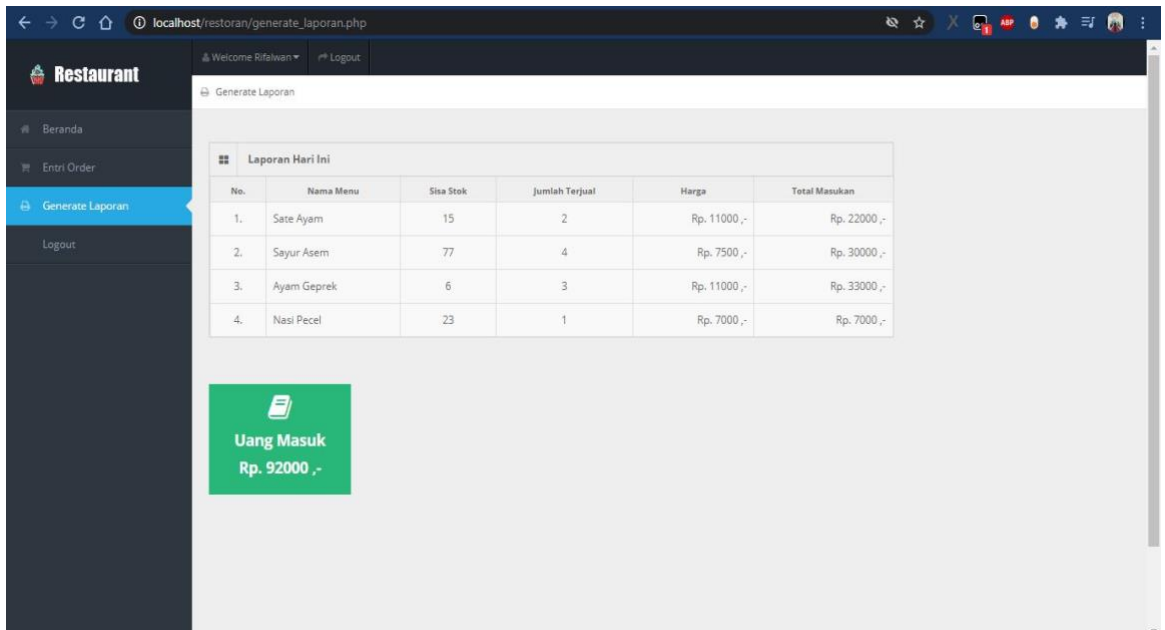
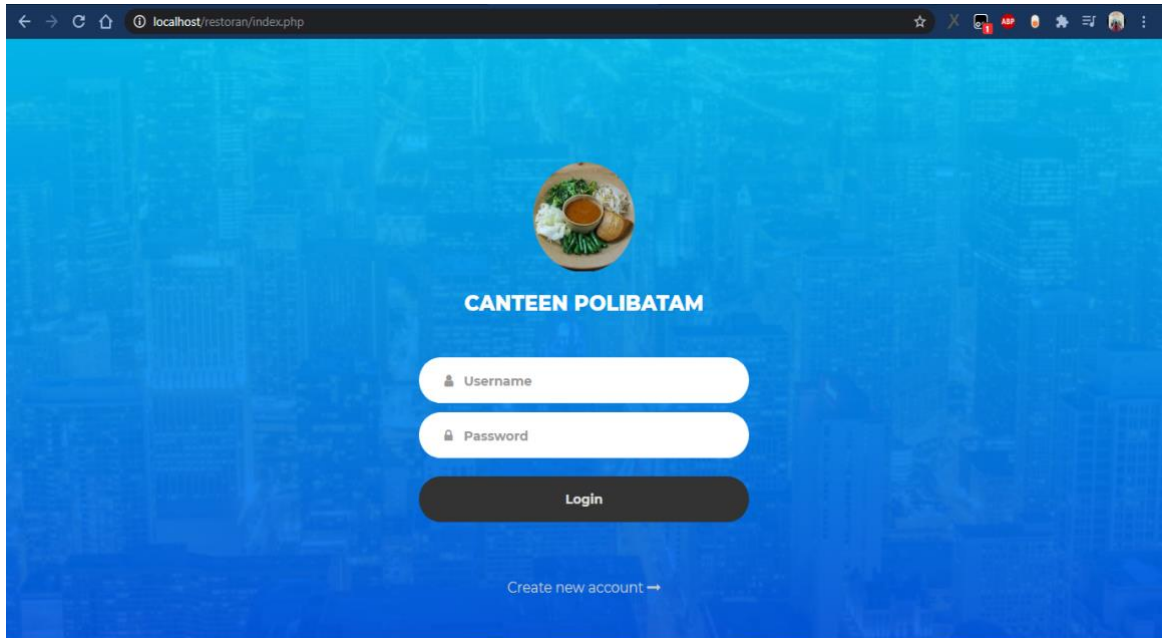
1. Parameter yang diuji : pembacaan sensor ultrasonik
2. Alat yang digunakan :
 - Sensor ultrasonik HC-SR04
 - Arduino UNO
 - laptop
3. Hasil :



4.4 Sistem Desain Web

4.4.1 Analisa Sistem Desain Web

1. Hasil :



BAGIAN 5: KENDALA DAN ISU

- Kendala yang pertama yaitu masalah koneksi Bluetooth. Dimana pada robot ini masi menggunakan koneksi Bluetooth sebagai komunikasinya. Semakin banyak koneksi Bluetooth di suatu ruangan semakin sulit juga untuk terhubung ke Bluetooth yang terdapat pada robot.
- Masalah sistem interface dashboard di indoor GPS. Dimana tampilan robot yang dipasang mobile beacon antara kondisi aktual dengan kondisi di software dashboardnya tidak sesuai

BAGIAN 6: KESIMPULAN

Berdasarkan hasil pengujian setiap masing masing anggota tim terkait tugas yang telah diberikan didapatkan kesimpulan sebagai berikut :

- Berdasarkan tujuan dari proyek ini sudah terlaksana beberapa tugas yang diberikan yaitu motor pada robot dapat berjalan sesuai perintah, Indoor GPS dapat Menampilkan posisi terakhir mobile beacon & robot di dashboard, Sensor sudah dapat mendeteksi halangan dan juga web sudah dapat diakses melalui localhost. Sedangkan untuk Sistem Auto Docking masih dalam tahap pembelajaran.

REFERENSI

Tuliskan sumber referensi yang digunakan dalam proyek.

1. Sistem lokalisasi robot Indoor GPS :
marvelmind.com (2020, 14 Desember). Marvelmind Indoor Navigation System Operating manual. Diakses pada 4 November 2020, dari
https://marvelmind.com/pics/marvelmind_navigation_system_manual.pdf.
marvelmind.com (2020, 13 Juli). Indoor “GPS” (with ± 2 cm precision) Placement Manual. Diakses pada 4 November 2020, dari
https://marvelmind.com/pics/Marvelmind_Robotics_ENG_placement_manual.pdf
marvelmind.com (2020, 11 Agustus). Inverse Architecture (IA). Diakses pada 4 November 2020, dari
https://marvelmind.com/pics/architectures_comparison.pdf
2. Sistem Deteksi halangan
Ardutech.com (2019, 17 Oktober). Pengukur Jarak dengan Arduino dan Sensor Ultrasonic. Diakses pada 5 Desember 2020, dari
<https://www.ardutech.com/pengukur-jarak-dengan-arduino-dan-sensor-ultrasonik/>

3. Sistem desain web

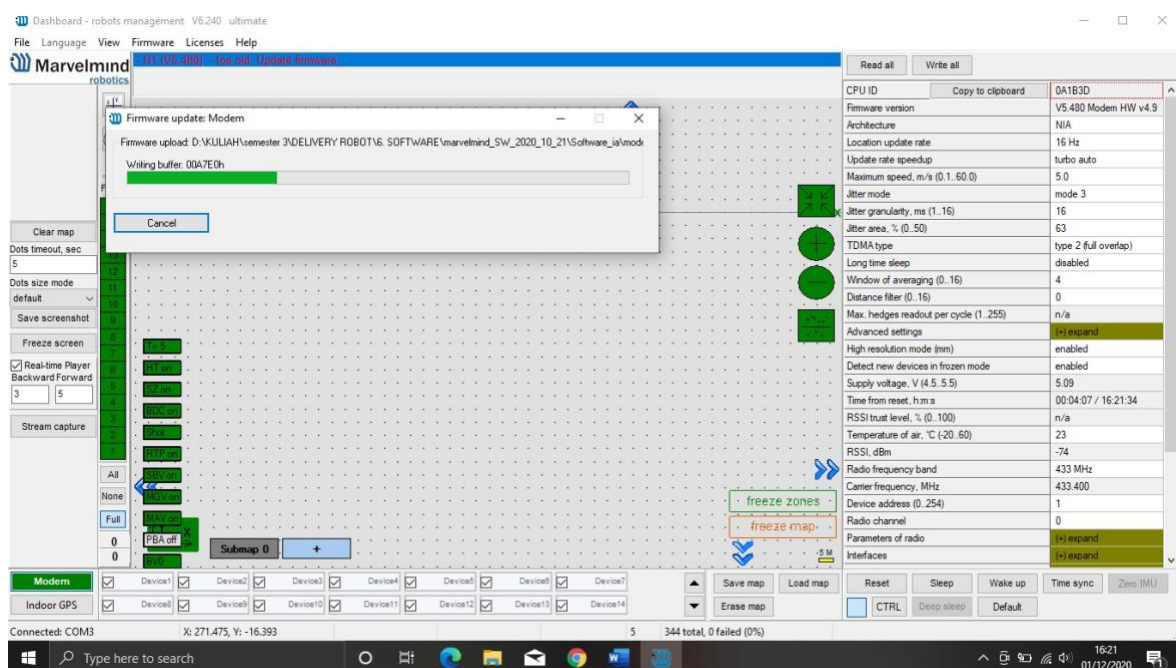
Jagat Koding. (2020, 4 Juni). Aplikasi Kasir Restoran dan Pelayanan Berbasis Web. Diakses pada 23 Januari 2021, dari

<https://www.youtube.com/watch?v=eJ4yix1-jtI&t=85s>

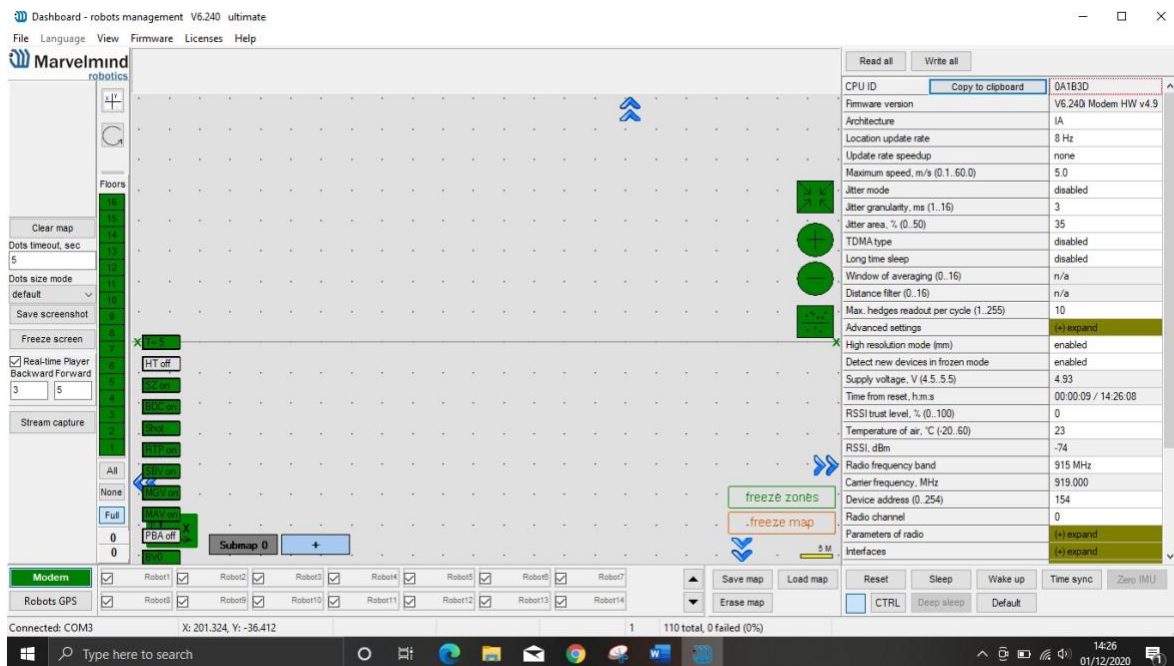
LAMPIRAN

PETUNJUK PENGGUNAAN

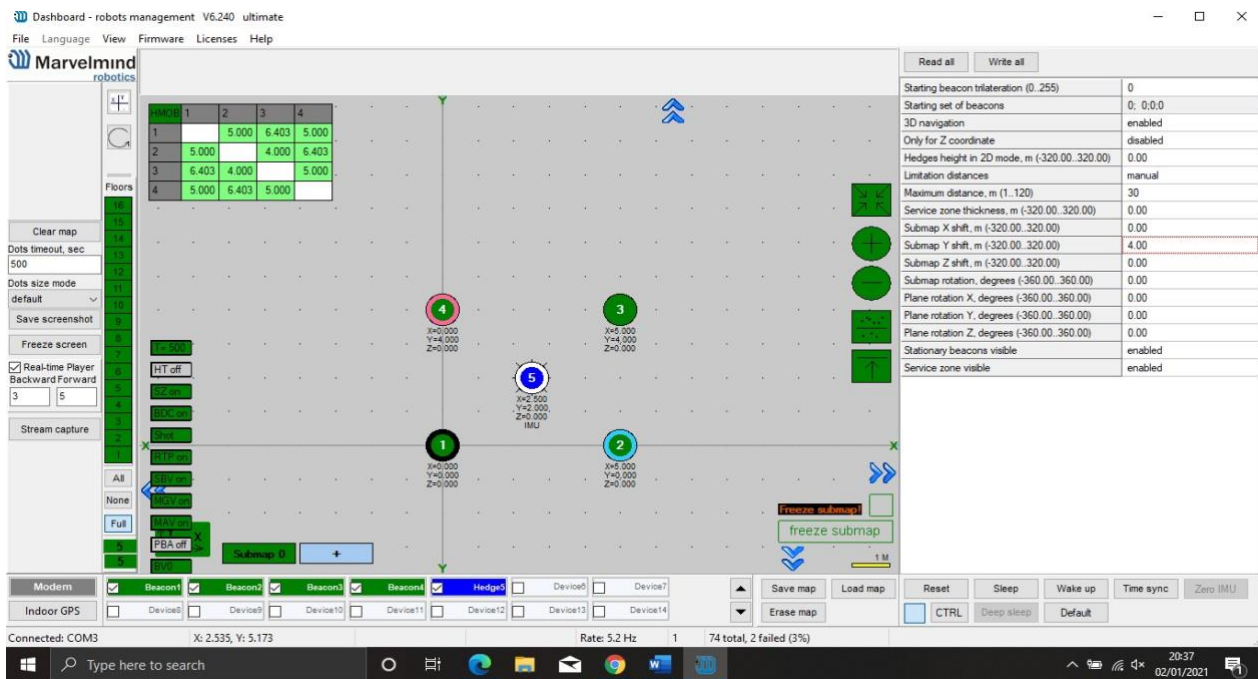
Instalasi sistem lokalisasi robot indoor GPS



Update firmware modem HW4.9

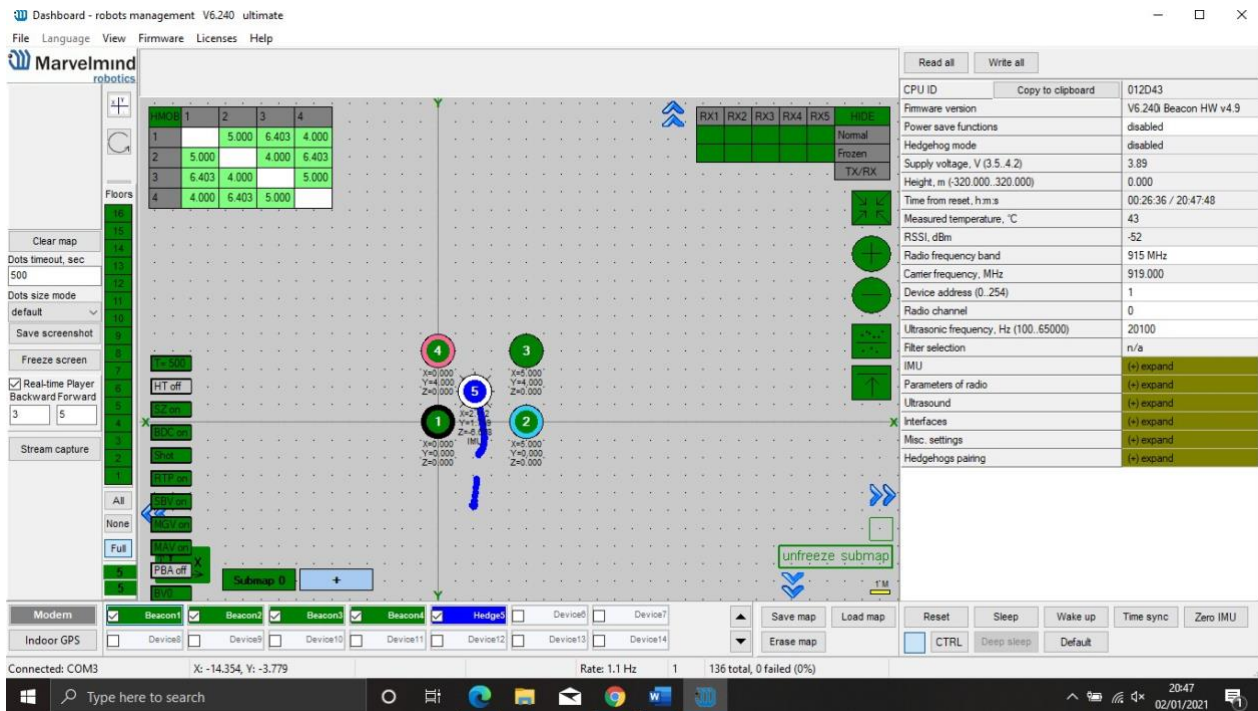


Modem sudah terbaca di dashboard software



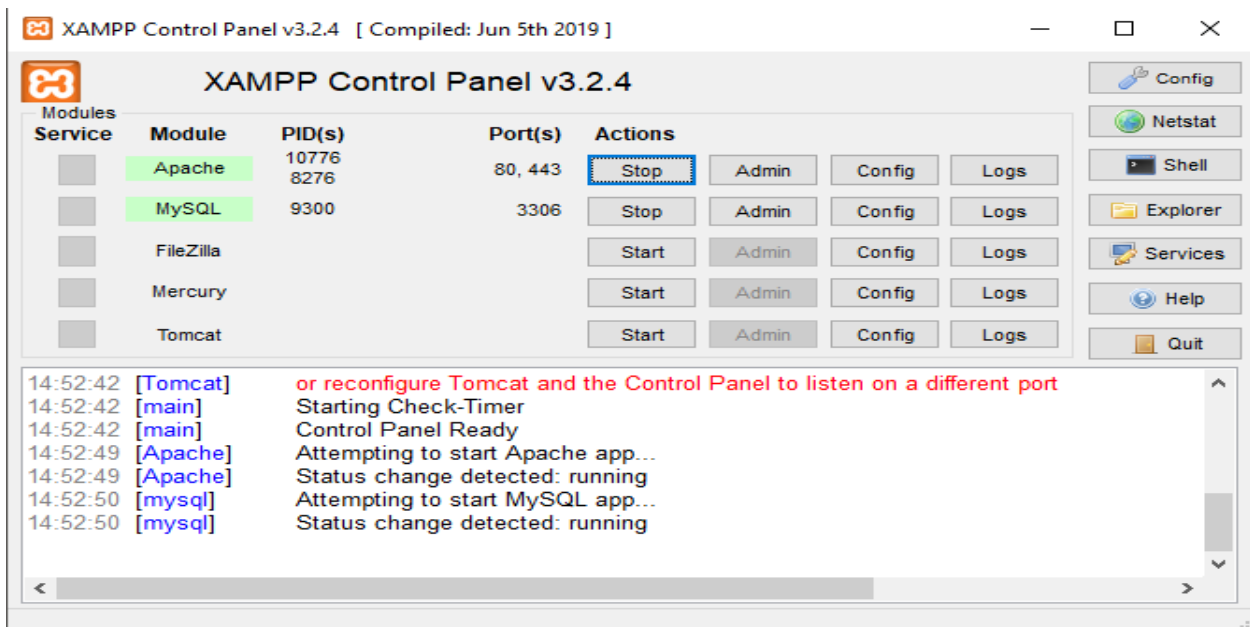
Tampilan beacon sudah muncul di dashboard

LAPORAN AKHIR PROYEK



Tampilan robot saat dijalankan

XAMPP WEB LOCALHOST APP



KODE PROGRAM**Kode Program pada Arduino Mega**

```

//Rahel Yulianti
//Robotika19
//Serial with parsing data
//delay with micros()
//delay with millis()
//encoder
//inverse kinematik
//BLUETOOTH Serial

#include<math.h>

//deklarasi pin Encoder 1
#define enA1 2
#define enB1 5
//deklarasi pin Encoder 2
#define enA2 3
#define enB2 4

//variable inverse kinematic differential wheels
float r = 120; //radius roda mm
float phi = 3.14;
float d = 340 ; // jarak antar roda mm
float v, w ; //Velocity mm/sec & omega rad/sec

```

```
float thetaDotL, thetaDotR; //kecepatan sudut roda kiri dan kanan (rps)
```

```
float Vinput, Winput; //variable tampung data dari serial w & v
```

```
float x1, y1, x2, y2;
```

```
//accel decceL
```

```
bool fungsinaik1 = false;
```

```
bool fungsiturun1 = true;
```

```
bool fungsinaik2 = false;
```

```
bool fungsiturun2 = true;
```

```
float y1ac, y2ac;
```

```
//ENCODER
```

```
int interval = 500;
```

```
bool state = false ;
```

```
int pulsa1, pulsa2;
```

```
int nilaipulsa1, nilaipulsa2;
```

```
float rps1, rpm1, rps2, rpm2;
```

```
// variable fungsi delay millis()
```

```
int State1; //deklarasi variable bertipe data integer
```

```
int State2;
```

```
long jumlahPulsa1, lastjumlahPulsa1, jumlahPulsa2, lastjumlahPulsa2; //deklarasi variable bertipe data long
```

```
unsigned long waktulama, waktusekarang, waktulama3, waktusekarang3; //deklrasi variable bertipe data  
unsigned long
```

```
unsigned long previousmicros1, previousmicros2, previousmicros3;
```

```

int interval1 = 100; //deklarasi variabel interval1 = periode waktu fungsi delay millis1 = 100 ms
int interval2 = 100; //deklarasi variabel interval2 = periode waktu fungsi delay millis2 = 100 ms
int interval3 = 50; //deklarasi variabel interval3 = periode waktu fungsi millis3 = 50 ms
bool parsing = false ; //deklarasi variable parsing bertipe data boolean
String sData, data[10]; //dekalarasi variable dData berupa string dan data[10] berupa string array 0 - 9 = 10

void setup() //void yang di jalankan sekali di awal pada saat arduino setup
{
    //set up nilai variable
    State1 = 0 ;
    State2 = 0 ;
    previousmicros1 = 0 ;
    previousmicros2 = 0 ;
    //menyatakan pin sebagai input yang di pull up
    pinMode(enA1, INPUT_PULLUP);
    pinMode(enB1, INPUT_PULLUP);
    pinMode(enA2, INPUT_PULLUP);
    pinMode(enB2, INPUT_PULLUP);
    //set up fungsi interrupt dengan mode change(rising dan falling)
    attachInterrupt(0, encoder1, CHANGE);
    attachInterrupt(1, encoder2, CHANGE);
    //set up baudrate pada serial
    Serial.begin(9600); //usb
    Serial1.begin(9600); //rightMotor
    Serial2.begin(9600); //leftMotor

```

```
Serial3.begin(9600); //bluetooth
```

```
}
```

```
void loop() { //fungsi yang akan diulang terus menerus
```

```
    unsigned long currentmicros2 = micros(); //deklarasi lokal variable untuk variable nilai micros
```

```
    if (currentmicros2 - previousmicros2 >= interval2) //jika nilai dari currentmicros2 - previousmicros2 >= interval2 maka menjalankan perintah didalam if
```

```
{
```

```
    previousmicros2 = currentmicros2; //update nilai dari previous micros menjadi nilai currentmicros sebelumnya
```

```
    Serial.print("m1>"); Serial.print(y1); Serial.print(","); Serial.print(y1ac); Serial.print(" | m2>");  
    Serial.print(y2); Serial.print(","); Serial.println(y2ac); //print nilai pmw dan accel&deccel
```

```
    //Serial.print("rps1= "); Serial.print(rps1); Serial.print(" rps2= "); Serial.println(rps2); //print nilai kecepatan m1&m2
```

```
}
```

```
//memanggil void ke dalam loop untuk dijalankan
```

```
parsingData();
```

```
INVERSE();
```

```
acceleration();
```

```
datasend();
```

```
hitungkecepatan();
```

```
}
```

```
void INVERSE() {
```

```
    v = Vinput; //velocity input cm/second
```

```
w = Winput; //input kecepatan sudut derajat/second
```

```
w = w * (phi / 180); // konversi dari derajat/second ke radian/second
```

```
thetaDotL = (1 / (2 * phi * r)) * (v - (w * d / 2)); //rumus kecepatan roda kanan (rps)
```

```
thetaDotR = (1 / (2 * phi * r)) * (v + (w * d / 2)); //rumus kecepatan roda kiri (rps)
```

```
x1 = abs(thetaDotL); //merubah nilai thetadotL menjadi nilai absolut
```

```
y1 = 107.8 * (pow(x1, -1.13)); //merubah nilai rps menjadi pwm menggunakan persamaan regresi
```

```
if (y1 > 10000) { //me rubah nilai yang besar dari 10000 menjadi 0
```

```
    y1 = 0 ;
```

```
}
```

```
else {
```

```
    y1 = y1 ;
```

```
}
```

```
x2 = abs(thetaDotR); //me rubah nilai yang besar dari 10000 menjadi 0
```

```
y2 = 107.8 * (pow(x2, -1.13));
```

```
if (y2 > 10000) {
```

```
    y2 = 0 ;
```

```
}
```

```
else {
```

```
    y2 = y2 ;
```

```
}
```

```

}

void datasend() {

    unsigned long currentmicros1 = micros(); //deklarasi lokal variable curentmicros1 dan memasukkan nilai
    dari fungsi micro second ke dalam variable curentmicros1

    if (currentmicros1 - previousmicros1 >= interval1) //jika nilai dari currentmicros1 - previousmicros1 >=
    interval1 maka menjalankan perintah didalam nya

    {

        Serial1.print("*"); Serial1.print(y1ac); Serial1.println("#"); //kirim data motor kanan berupa pwm melalu
        serial1

        Serial2.print("*"); Serial2.print(y2ac); Serial2.println("#"); //kirim data motor kanan berupa pwm melalu
        serial2

        previousmicros1 = currentmicros1; //update nilai dari previous micros menjadi nilai currentmicros
        sebelumnya

    }

}

void parsingData() {

    //change to serial 3 to use bluetooth

    while (Serial3.available()) { //ketika serial3 menerima data maka akan menjalankan perintah yang ada di
    dalam while

        char inChar = Serial3.read(); //membaca data serial3 dan memasukkan kedalam variable inChar yang
        bertipe data char

        sData += inChar; //memasukkan data inchar ke dalam varibale char

        if (inChar == '#') { //jika inchar berupa char # maka parsing = true

            parsing = true;

        }

    }

```

```

if (parsing == true ) {    //jika parsing == true maka akan menjalankan perintah di dalam if

    int q = 0;              //deklarasi lokal variable q = 0

    for (int i = 0; i <= sData.length(); i++) {          //jika nilai i <= s.data maka akan di cek setiap
character
        if (sData[i] == '*' || sData[i] == ',' || sData[i] == '#') { //mengecek karakter pada masukan '*' ',' '#'
maka akan menambah nilai variable q

            q++;

            data[q] = "";

        }

        else {

            data[q] += sData[i];          //jika inputan berisi selain '*' ',' '#' maka akan di masukkan ke
dalam data[q]

            //data[q].toInt();

        }

    }

    //konversi data[q] yang berupa string menjadi integer lalu memasukkan ke dalam variabel

    Vinput = data[1].toInt();

    Winput = data[2].toInt();

    //jika nilai i > sData.lenght maka bool parsing false dan string sData = ""

    parsing = false;

    sData = "";

}

}

}

void encoder1() { //ENCODER1 Left

```

```

if (digitalRead(enA1) == digitalRead(enB1)) //jika nilai enA1 sama dengan nilai enB1 maka pulsa decrement
{
    pulsa1--;
}
else //jika tidak maka nilai nya increment
{
    pulsa1++;
}
jumlahPulsa1 = pulsa1 / 2.5;
}

void encoder2() //ENCODER2 Right
{
    if (digitalRead(enA2) == digitalRead(enB2)) //jika nilai enA2 sama dengan nilai enB2 maka pulsa increment
    {
        pulsa2++;
    }
    else //jika tidak maka nilai nya decrement
    {
        pulsa2--;
    }
    jumlahPulsa2 = pulsa2 / 2.5;
}

void hitungkecepatan()

```

```

{
    //jumlah pulsa satu putaran = 823.00 (pulsa encoder)

    waktusekarang = millis();          //memasukan nilai dari millis() ke dalam variable waktusekarang

    if (waktusekarang - waktulama >= interval)    //jika nilai waktusekarang - waktulama >= interval maka
    akan menjalankan perintah di dalam if

    {
        rps1 = (jumlahPulsa1 / 823.00) / 0.5;    //hitung rps
        rps2 = (jumlahPulsa2 / 823.00) / 0.5;
        rpm1 = rps1 / 60;                        //hitung rpm
        rpm2 = rps2 / 60;
        //reset nilai
        pulsa1 = 0;
        pulsa2 = 0;
        jumlahPulsa1 = 0 ;
        jumlahPulsa2 = 0 ;
        waktulama = waktusekarang;
    }
}

void acceleration() {
    // cek nilai accel
    if (y1ac > 2000) {
        y1ac = 0;
    }
    else {

```

```

y1ac = y1ac;
}
if (y2ac > 2000) {
    y2ac = 0 ;
}
else {
    y2ac = y2ac ;
}

waktusekarang3 = millis();
if (waktusekarang3 - waktulama3 >= interval3)
{
    waktulama3 = waktusekarang3;

    //motor1 accel decel

    if (y1 > 0) {          //jika nilai y1 > 0 maka akan masuk ke if

        if (y1ac < 1) {      //jika nilai y1ac < 1 maka y1ac = 2000
            y1ac = 2000;
        }
        else {
            if (y1ac > y1) {    //jika nilai y1ac > 1 maka akan di decremental sampai nilai mendekati y1
                y1ac -= y1ac * 0.2;  //ACCELERATION DECREMENTAL 20% MOTOR 1
            }
        }
    }
}

```

```

}

else {

    if (y1ac > 0) {

        y1ac += y1ac * 0.2;    //DECCELERATION INCREMENTAL 20% MOTOR 1

    }

}

//motor2 accel decel

if (y2 > 0) {

    if (y2ac < 1) {

        y2ac = 2000;

    }

    else {

        if (y2ac > y2) {

            y2ac -= y2ac * 0.2;    //ACCELERATION DECREMENTAL 20% MOTOR 2

        }

    }

}

else {

    if (y2ac > 0) {

        y2ac += y2ac * 0.2;    //DECCELERATION INCREMENTAL 20% MOTOR 2

    }

}

}

}

```

Kode Program pada Arduino UNO

```
//Rahel Yulianti
```

```
//Robotics19
```

```
bool parsing = false; //deklarasi variable parsing dengan tipe data boolean dengan nilai awal false
```

```
String sData, data[10]; //deklarasi variable sData dengan tipe data string dan data dengan tipe data String array
```

```
#define puls 2 //deklarasi pin digital 2 arduino sebagai puls pin
```

```
#define LED 13 //deklarasi pin digital 13 arduino sebagai LED pin
```

```
float Interval; //deklarasi variable Interval dengan tipe data Float
```

```
void setup() {
```

```
    // put your setup code here, to run once:
```

```
    Interval = 0.00 ; //setup nilai Interval dengan nilai 0.00
```

```
    pinMode(LED, OUTPUT); //setup pin LED dengan mode output
```

```
    pinMode(puls, OUTPUT); // setup pin puls dengan Mode output
```

```
    Serial.begin(9600); // setup serial dengan baudrate 9600
```

```
}
```

```
void loop() {
```

```
    if (Interval != 0 && Interval > 20) { //jika nilai interval tidak = 0 dan besar dari 20 maka akan menjalankan perintah yang ada dalam if
```

```
        digitalWrite(puls, HIGH); // arduino memberika output HIGH pada pin puls
```

```
        delayMicroseconds(Interval); // delay microsecond dengan nilai delay (Interval)
```

```
        digitalWrite(puls, LOW); // arduino memberika output LOW pada pin puls
```

```

delayMicroseconds(Interval); // delay microsecond dengan nilai delay (Interval)

else {

    digitalWrite(puls, LOW); //jika tidak maka arduino akan memberikan output LOW pada pin puls
}

// put your main code here, to run repeatedly:

while (Serial.available()) { //ketika serial menerima data maka akan menjalankan perintah yang ada di
    dalam while

        char inChar = Serial.read(); //membaca data serial dan memasukkan kedalam variable inChar yang
        bertipe data char

        sData += inChar; //memasukkan data inchar ke dalam varibale char

        if (inChar == '#') { //jika inchar berupa char # maka parsing = true

            parsing = true;

        }

        if (parsing) { //jika parsing == true maka akan menajalankan perintah di dalam if

            int q = 0; //deklarasi lokal variable q = 0

            for (int i = 0; i <= sData.length(); i++) { //jika nilai i <= s.data maka akan di cek setiap
            character

                if (sData[i] == '*' || sData[i] == ',' || sData[i] == '#') { //mengecek karakter pada masukan '*' ',' '#'
                maka akan menambah nilai variable q

                    q++;

                    data[q] = "";

                }

                else {

                    data[q] += sData[i]; //jika inputan berisi selain '*' ',' '#' maka akan di masukkan
                    ke dalam data[q]

                    //data[q].toInt();

```

```
}  
  
}  
  
//konversi data[q] yang berupa string menjadi integer lalu memasukkan ke dalam variabel  
Interval = data[1].toFloat();  
  
//jika nilai i > sData.lenght maka bool parsing false dan string sData = ""  
  
parsing = false;  
  
sData = "";  
  
  
}  
  
}  
  
//perintah lain  
  
}
```

Sensor ultrasonic

```

unsigned long myTime;

int trigPin1 = 13;

int echoPin1 = 12;

int trigPin2 = 10;

int echoPin2 = 9;

int trigPin3 = 8;

int echoPin3 = 7;

long duration1, distance1;

long duration2, distance2;

long duration3, distance3;

void setup() {

  // put your setup code here, to run once:

  Serial.begin (9600);

  pinMode(trigPin1, OUTPUT);

  pinMode(echoPin1, INPUT);

  pinMode(trigPin2, OUTPUT);

  pinMode(echoPin2, INPUT);

  pinMode(trigPin3, OUTPUT);

  pinMode(echoPin3, INPUT);

void Print() {

  long waktusekarang;

  long interval1 = 500;

  long waktuberikutnya;

```

```
waktusekarang = millis();

if (waktusekarang - waktuberikutnya >= interval1) {

    waktuberikutnya = waktusekarang;

    if (distance1 >= 1000 && distance1 <= 0) {

        Serial.print(">>Out of range");

    }

    else {

        Serial.print ( ">>Sensor1 ");

        Serial.print ( distance1);

        Serial.print("cm");

    }

    if (distance2 >= 1000 && distance2 <= 0) {

        Serial.print("Out of range");

    }

    else {

        Serial.print(">>Sensor2 ");

        Serial.print(distance2);

        Serial.print("cm");

    }

    if (distance3 >= 1000 && distance3 <= 0) {

        Serial.print(">>Out of range");

    }

}
```

```

else {
    Serial.print(" >>Sensor3 ");
    Serial.print(distance3);
    Serial.println("cm");
}
}
}

void loop() {
    // put your main code here, to run repeatedly:
    Print();

    digitalWrite(trigPin1, LOW); // Added this line
    delayMicroseconds(2); // Added this line
    digitalWrite(trigPin1, HIGH);
    delayMicroseconds(10); // Added this line
    digitalWrite(trigPin1, LOW);
    duration1 = pulseIn(echoPin1, HIGH);
    distance1 = (duration1 / 2) / 29.1;
    myTime = millis();

    digitalWrite(trigPin2, LOW); // Added this line
    delayMicroseconds(2); // Added this line
    digitalWrite(trigPin2, HIGH);
    delayMicroseconds(10); // Added this line
    digitalWrite(trigPin2, LOW);

```

```
duration2 = pulseIn(echoPin2, HIGH);
```

```
distance2 = (duration2 / 2) / 29.1;
```

```
myTime = millis();
```

```
digitalWrite(trigPin3, LOW); // Added this line
```

```
delayMicroseconds(2); // Added this line
```

```
digitalWrite(trigPin3, HIGH);
```

```
delayMicroseconds(10); // Added this line
```

```
digitalWrite(trigPin3, LOW);
```

```
duration3 = pulseIn(echoPin3, HIGH);
```

```
distance3 = (duration3 / 2) / 29.1;
```

```
myTime = millis();
```

```
}
```